AD-A186 583   SUPERCOMPUTERS FOR SOLVING PDE (PARTIAL DIFFERENTIAL   1/1
               EQUATIONS) PROBLEMS(U) UNIVERSITY OF SOUTHERN
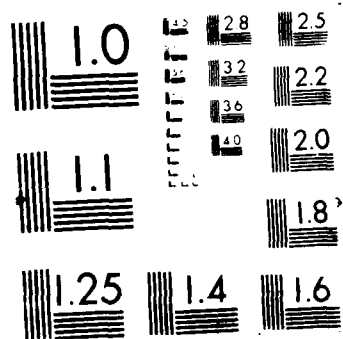   CALIFORNIA LOS ANGELES COMPUTER RESEAR   K HWANG
UNCLASSIFIED   11 AUG 87 AFOSR-TR-87-1275 AFOSR-86-0088   F/G 12/6   NL

AD-A186 583

REPORT DOCUMENTATION PAGE

DTIC FILE COPY ②

1a. REPORT SECURITY CLASSIFICATION	1b RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	Approved for public release; distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
	AFOSR·TR· 87 - 1 2 7 5

6a. NAME OF PERFORMING ORGANIZATION Univ. of Southern California	6b OFFICE SYMBOL (If applicable)	7a NAME OF MONITORING ORGANIZATION AFOSR/NM
6c. ADDRESS (City, State, and ZIP Code) Los Angles, CA 90089-0781		7b ADDRESS (City, State, and ZIP Code) AFOSR/NM Bldg 410 Bolling AFB DC 20332-6448

8a. NAME OF FUNDING/SPONSORING ORGANIZATION AFOSR	8b OFFICE SYMBOL (If applicable) NM	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER AFOSR 86-0008

8c. ADDRESS (City, State, and ZIP Code) AFOSR/NM Bldg 410 Bolling AFB DC 20332-6448	10 SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO. 61102F	PROJECT NO. 2304	TASK NO. A3	WORK UNIT ACCESSION NO.

11 TITLE (Include Security Classification)

Supercomputers for Solving PDE Problems

12. PERSONAL AUTHOR(S)   Dr. Kai Hwang

13a. TYPE OF REPORT Final	13b. TIME COVERED FROM 9/20/87 TO 8/11/87	14. DATE OF REPORT (Year, Month, Day) 8/11/87	15. PAGE COUNT

16. SUPPLEMENTARY NOTATION

DTIC ELECTE OCT 1 3 1987

17.	COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This project investigated parallel/vector supercomputer architectures for solving Air Force problems, which demand the solution of partial differential equations (PDEs). We have developed an underline orthogonal multiprocessor (OMP) architecture for efficiently implementing the SLOR and ADI methods in solving PDEs.

Another parallel PDE machine architecture, called the V-tree multiprocessor, has been developed for mapping the multigrid algorithms. This V-tree is shown to be more effective than the well-known hypercube and mesh architectures.

Both the OMP and the V-tree architectures can demonstrate linear speedup by exploiting parallelism and vectorization. Continued efforts are needed to expand these initial studies into real hardware experiments and software simulations to verify the theoretical predictions on speedup performance.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL Maj. John P. Thomas	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL NM

DD FORM 1473, 84 MAR   83 APR edition may be used until exhausted.   SECURITY CLASSIFICATION OF THIS PAGE

All other editions are obsolete.

Annual Report of Research Progress on
Supercomputers for Solving PDE Problems *

Dr. Kai Hwang
Computer Research Institute
University of Southern California
Los Angeles, CA 90089

Our research achievements in this reporting period include several related research topics. We have investigated the mapping of PDE algorithms onto various multi-processor architectures [1,2]. A language construct for developing parallel programs is proposed [3]. Other efforts include the domain decomposition approach to solving PDE problems [4] and efficient preprocessing and postprocessing in finite element analysis [5]. All the work was aimed at boosting the speed at which large-scale PDE problems can be solved via the use of parallel computers.

Mapping of parallel algorithms for solving PDE problems onto *Orthogonal Multiprocessor* (OMP), a multiprocessor architecture conceived at USC, has been investigated in depth. Specifically, two methods, SLOR and ADI, are mapped onto the architecture. Analytical results show that linear speedup with the number of processors can be achieved by a proper distribution of data in the memory modules.

Mapping of multigrid algorithms is examined in the context of four classes of multicomputer architectures, namely, *trees*, *hypercubes*, *meshes*, and the *OMP*. Different mapping strategies are presented and analyzed in terms of load balance achieved and communication penalty paid in each case. Extensive comparisons have been conducted to provide useful guidelines in the selection of suitable mapping strategies for different architectures.

Molecule language is proposed to bridge the gap between the development of hardware and software supports for parallel computers. It provides syntax and semantics rules which allow the user to specify the desired computation modes that best match problem characteristics. Such a concurrent language approach is instrumental to the effective solution of PDE problems on supercomputers.

Other related research results have also been reported on further development and potential optical implementation of pipeline nets [6] used in the Remps architecture [7]. Trends of parallel processing, including recent advances in optical and neural computing, and their prospective applications to PDE solutions are summarized in [8].

Research Team

Principal Investigators: Dr. Kai Hwang
Professor of Electrical Engineering and Computer Science

Dr. W. Proskurowski
Associate Professor of Mathematics

Ph.D. Students: A. Louri, D. Kim, Z. Xu, and H.C. Wang

Relevant Publications

[1] K. Hwang, P. Tseng, and D. Kim, "An Orthogonal Multiprocessor for Efficient Parallel Processing," *IEEE Transactions on Computers*, to appear 1988.

[2] K. Hwang and H. Wang, "Partitioning of Multigrid Algorithms for Multiprocessor Supercomputers," *Tech. Rep. CRI 87-20*, University of Southern California, August, 1987.

[3] Z. Xu and K. Hwang, "Molecule: A Language Construct for Layered Development of Parallel Programs," *IEEE Transactions on Software Engineering*, to appear 1988.

[4] W. Proskurowski, M. Dryja, and O. Widlund, "Numerical Experiments and Implementations of a Domain Decomposition Method," *Tech. Rep. CRI 86-37*, University of Southern California, December, 1986.

[5] P. Tseng and K. Hwang, "Parallel Preprocessing and Postprocessing in Finite Element Analysis on a Multiprocessor Computer," in *ACM/IEEE 1986 Fall Joint Computer Conference*, pp. 307–314, November, 1986.

[6] K. Hwang and Z. Xu, "Pipeline Nets for Compound Vector Supercomputing," *IEEE Transactions on Computers*, January, 1988.

[7] K. Hwang, Z. Xu, and A. Louri, "Remps: An Electro-Optical Supercomputer for Parallel Solution of PDE problems," in *Proc. of the Second International Conference on Supercomputing*, pp. 301–310, May, 1987.

[8] K. Hwang, "Advanced Parallel Processing and Supercomputer Architectures," *Proceedings of the IEEE*, October, 1987.

Research Project:
Supercomputers for Solving PDE Problems

Principal Investigator:
Dr. Kai Hwang
University of Southern California

- An Orthogonal Multiprocessor (OMP) architecture is developed for solving PDE problems using the SLOR and ADI methods.

- Linear speedup can be achieved with OMP architecture, on which the SLOR and ADI methods are partitioned for parallel processing.

- A V-Tree Multiprocessor is suggested for parallel implementation of the V-cycle in multigrid algorithms.

The OMP Architecture and Orthogonal Memory Accesses:

- The Bus Controller enables either row memory accesses (using the row buses) or column memory accesses (using the column buses) but not both at the same time.

- These orthogonal memory access patterns avoid conflicts completely and, therefore, achieve full memory bandwidth.
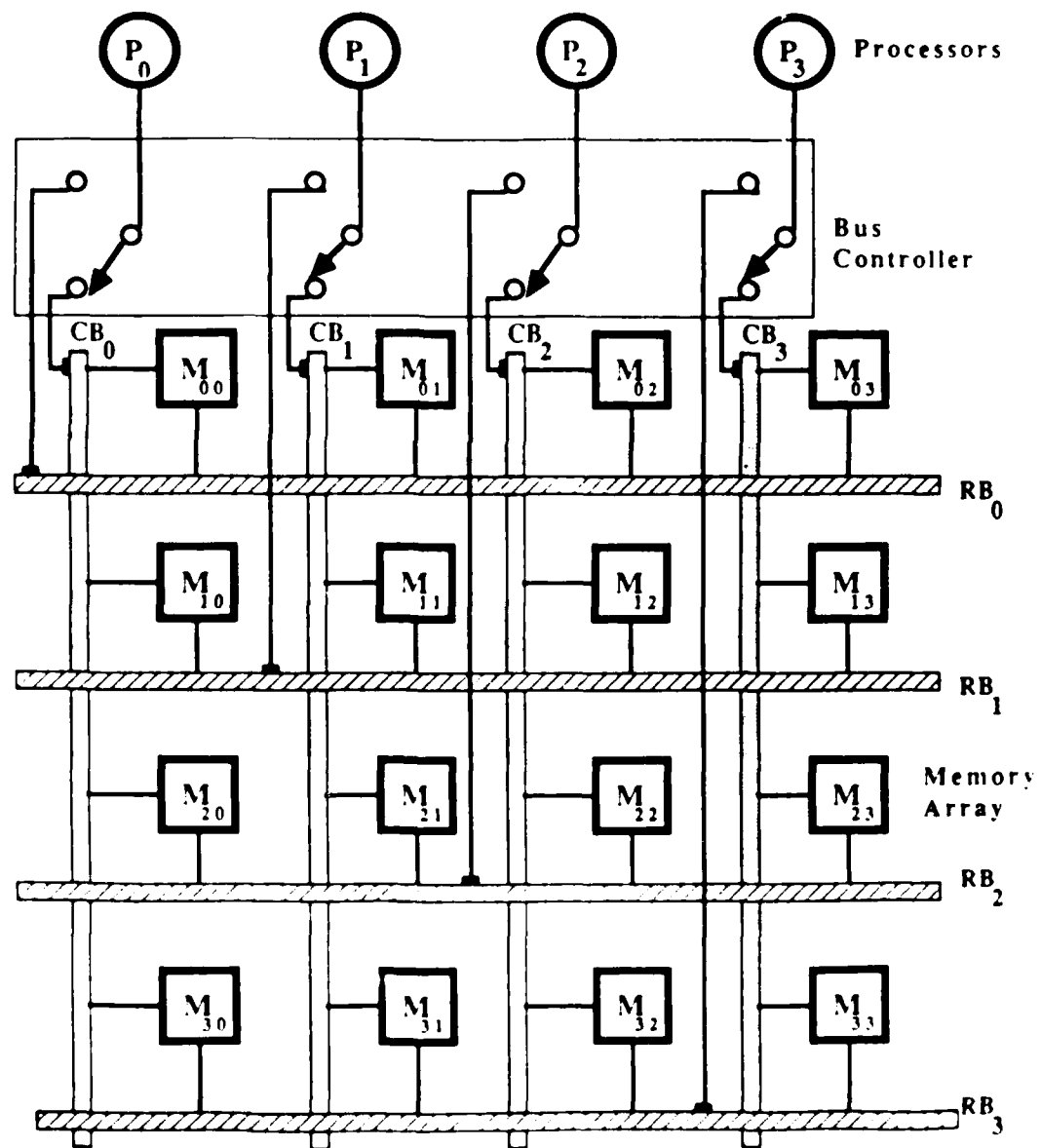

Figure 1. A 4-processor Orthogonal Multiprocessor architecture

The SLOR Method on OMP:

- The grid points are evenly distributed into two subsets by alternate lines (Fig. 2).

- Each iteration requires $O(k^2/n)$ time on an OMP with n processors, where $k*k$ is the grid size. Note that the same problem requires $O(k^2)$ time on a uniprocessor system.
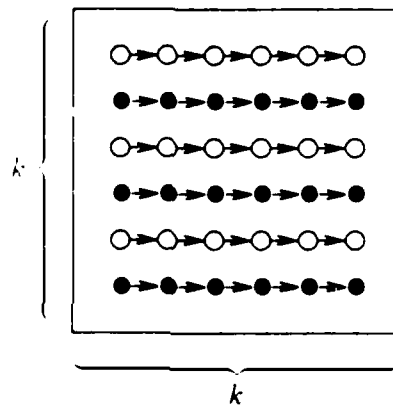

Figure 2. The row distribution for either the SLOR method or ADI method

The ADI Method on OMP:

- The grid points are distributed to the row memory and column memory (Fig. 2 and Fig. 3).

- Each iteration of the ADI method on a grid of $k*k$ points can be done in $O(k^2/n)$ time on an OMP with n processors. A linear speedup is achieved compared with a uniprocessor.
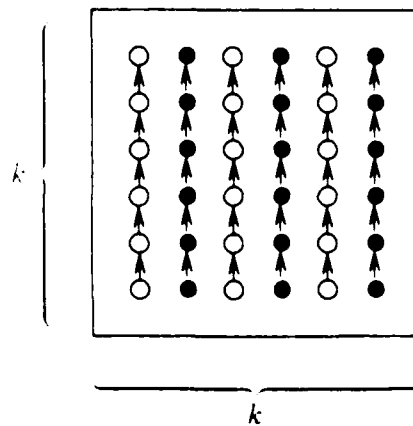

Figure 3. The column distribution for the ADI method

Implementing Multigrid Algorithms on a V-tree Architecture:

- Parallelization of the V-cycle in a multigrid algorithm

- Efficient implementation of multigrid algorithms on a V-tree multiprocessor system
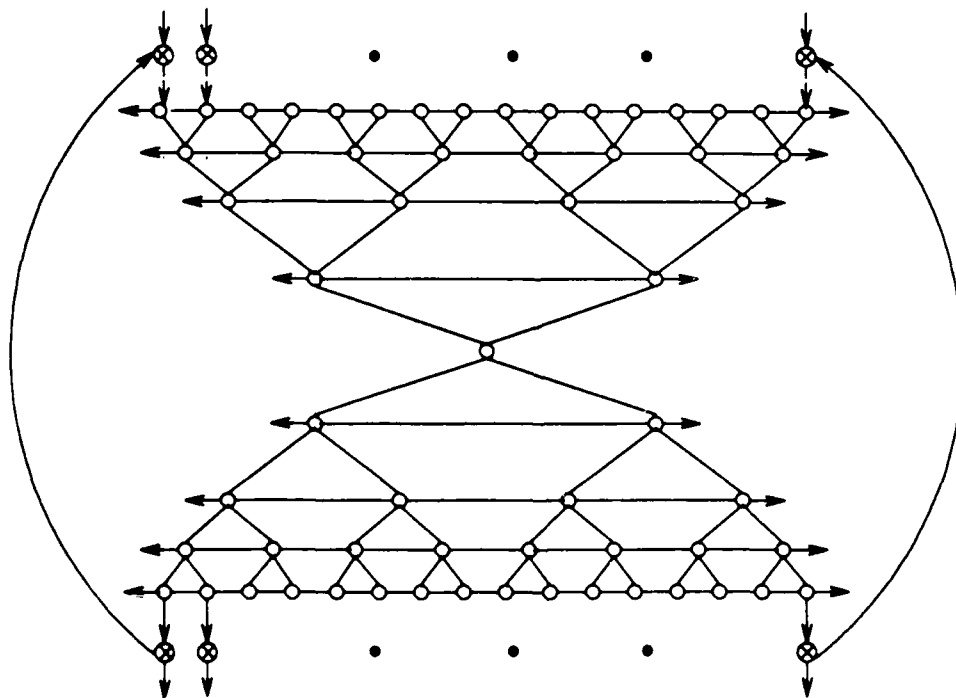

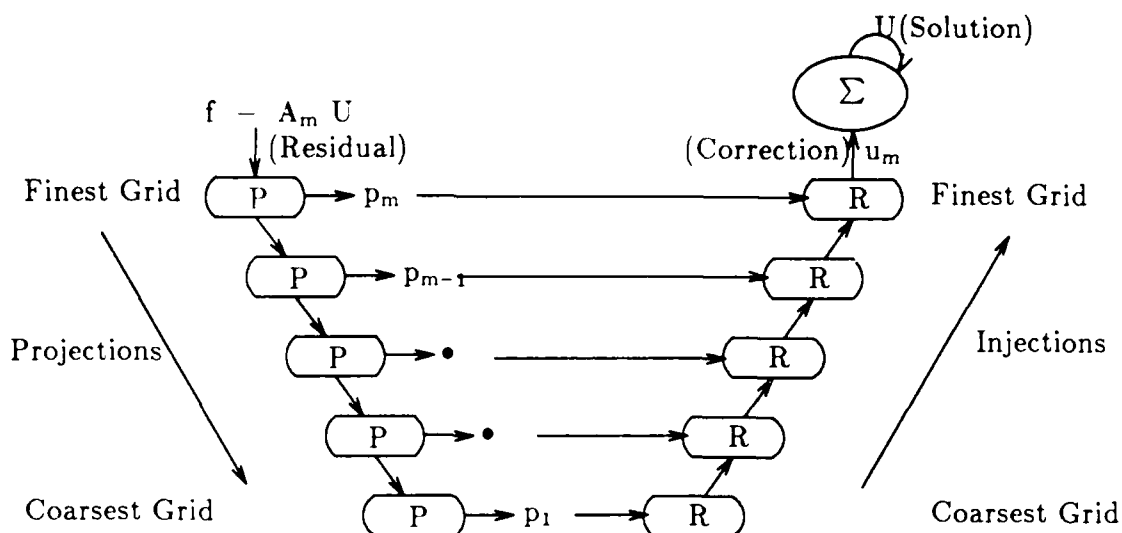Figure 6. The architecture of a V-tree architecture constructed from two augmented trees joined at the roots

Figure 4. A sequential multigrid algorithm has a V-cycle of successive projections from fine to coarse grids and a sequence of injections in the reverse direction
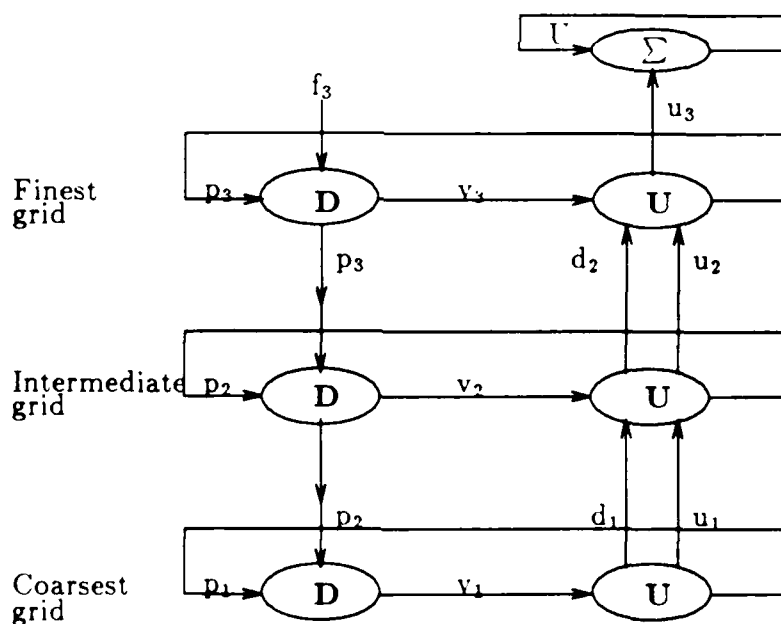

Figure 5. Concurrent multigrid algorithm

- One tree is devoted to the projection sequence on the V-cycle and the other to the injection sequence.

- Both parallelism and vectorization are exploited on the V-tree.

- Higher throughput and better processor utilization are achieved on the V-tree.

END

DATE

FILM

JAN

1988